

WHAT IS CLAIMED IS:

*Sent
a*) 1. A data processing system stored on a computer-readable medium comprising:
one or more event modules including code that generates an event data signal representative of a particular event;
5 one or more scripts each of said one or more scripts having one or more instructions;
one or more processing modules each including code that provides processed data to said one or more scripts; and
a task module, selectively communicating with each of
10 said one or more event modules, including code for execution of a selected one of said one or more scripts that corresponds to said event data signal;
wherein during said execution, said selected script interfaces with one or more of said one or more processing
15 modules and incorporates results of said interfacing into said one or more instructions of said selected script.

2. The system as claimed in claim 1, wherein said task module executes a plurality of said one or more scripts substantially simultaneously.

sac
B2

3. The system as claimed in claim 2 further comprising:
a converter module, in communication with said task
module, including code that maps said event data signal to one
or more of said plurality of said one or more scripts upon
reception of said event data signal by said task module.

4. The system as claimed in claim 1 wherein said one or more
processing modules provide event data signals, representative
of a particular event, to said task module.

5. The system as claimed in claim 1 further comprising:
a status monitoring module, in communication with said
task module, including code that provides information to said
task module relating to operating conditions of said one or
more processing modules.

6. The system as claimed in claim 5 wherein said status
monitoring module is in direct communication said with said
one or more processing modules.

sac
c3

7. The system as claimed in claim 5 wherein during said
execution of said selected script, said status monitoring
module stores data associated with said selected script in an
associated memory.

*sab
b3*

8. The system as claimed in claim 1 further comprising:
a load balancing module, in communication with said task
module, including code that dynamically selects available ones
of said one or more processing modules to perform processing.

9. The system as claimed in claim 8 wherein said load
balancing module is in direct communication with said one or
more processing modules.

10. The system as claimed in claim 1 wherein said task module
interfaces with said one or more processing modules for bi-
directionally and substantially simultaneously transmitting
data between said one or more processing modules and said task
module.

5

11. The system as claimed in claim 1 further comprising:
a resource management module, in communication with said
task module, including code that dynamically assigns
processing functions to said one or more processing modules.

12. The system as claimed in claim 11 wherein said resource
management module is in direct communication with said one or
more processing modules.

00000000000000000000000000000000

13. The system as claimed in claim 1 further comprising:
one or more initiator modules including code that
provides a communication interface between an associated one
of said one or more processing modules and said task module.

14. The system as claimed in claim 13 wherein each of said
one or more initiator modules communicates with said
associated one of said one or more processing modules
regardless of native applications contained on said associated
5 one of said one or more processing modules.

15. The system as claimed in claim 13 further comprising:
a protocol disposed between each of said one or more
initiator modules and said task module for providing a
communication interface therebetween.

16. The system as claimed in claim 13 further comprising:
a protocol disposed between each of said one or more
initiator modules and said associated one of said one or more
processing modules for providing a communication interface
5 therebetween.

17. The system as claimed in claim 1 further comprising:
one or more client modules including code that provides a
communication interface between an associated one of said one
or more event modules and said task module.

18. The system as described in claim 17 further comprising:
a protocol disposed between said task module and each of
said one or more client modules for providing a communication
interface therebetween.

19. The system as claimed in claim 17 further comprising:
a protocol disposed between each of said one or more
client modules and said associated one of said one or more
event modules for providing a communication interface
therebetween.

20. The system as claimed in claim 1 wherein each of said one
or more scripts is preprogrammed to iteratively update its
contents.

21. The system as claimed in claim 1 further comprising:
a storage module, in communication with said task module,
for providing storage for said system.

Sel
22. The system as claimed in claim 21 wherein said storage module comprises a computer-readable medium.

23. The system as claimed in claim 22 wherein said computer readable medium comprises a persistent memory.

*Sel
B4*
24. The system as claimed in claim 21 further comprising:
a script building module, in communication with said storage module, including code that creates said one or more scripts.

*Sel
B5*
25. The system as claimed in claim 24 wherein said script building module includes a standard language interface.

26. The system as claimed in claim 24 wherein said script building module includes a graphical user interface.

*Sel
B5*
27. The system as claimed in claim 24 wherein said script building module dynamically updates and modifies said one or more scripts.

28. The system as claimed in claim 1 further comprising:
a protocol for providing a communication interface between said task module and each of said one or more event modules.

29. The system as claimed in claim 1 further comprising:
a protocol for providing a communication interface
between said task module and each of said one or more
processing modules.

Sayl A² 30. The system as claimed in claim 1 further comprising:
a responder module, in communication with said task
module, including code that transmits response data, resulting
from said execution, from said task module to particular said
one or more processing modules or to particular said one or
more event modules.

Sayl B¹ 31. The system as claimed in claim 1 further comprising:
an administrative module, in communication with said task
module, including code that receives and presents data that
relates to said one or more processing modules.

32. The system as claimed in claim 1 further comprising:
one or more application peripherals in communication with
an associated one of said one or more processing modules or an
associated one of said one or more event modules.

Sul
33. A data processing system stored on a computer-readable medium comprising:

one or more event modules each including code that generates an event data signal representative of a particular
5 event;

one or more scripts each of said one or more scripts having one or more instructions;

one or more processing modules each including code that provides said one or more scripts with processed data;

10 a task module, selectively communicating with each of said one or more event modules, including code for execution of a selected one of said one or more scripts that correspond to said event data signal;

wherein during said execution, said selected script
15 interfaces with one or more of said one or more processing modules and incorporates results of said interfacing into said one or more instructions of said selected script;

a resource management module in direct communication with said one or more event modules, said task module and said one
20 or more processing modules, including code that dynamically assigns processing functions to said one or more processing modules; and

an administrative module in direct communication with said task module, said one or more event modules, said one or
25 more processing modules and said resource management module,

including code that receives and presents data relating to said one or more processing modules.

34. A method of data processing comprising the steps of:
generating at least one event data signal at one or more peripheral modules;

5 selected script chosen from one or more scripts, each said one or more scripts having one or more instructions for performing data gathering steps; and

executing, by a task module, said selected script to gather processed data from one or more of said one or more 10 peripheral modules;

wherein during said executing step, said selected script dynamically incorporates results of said executing into said one or more instructions of said selected script.

35. The method as claimed in claim 34 wherein said one or more peripheral modules and said task module communicate via a communication interface.

36. The method as claimed in claim 34 further comprising the step of:

dynamically managing operating functions of said one or more peripheral modules.

See C10

37. The method as claimed in claim 34 further comprising the steps of:

producing response data signals as a result of said executing step; and

5 transmitting said response data signals from said task module to selected said one or more peripheral modules.

38. The method as claimed in claim 37 further comprising the step of:

translating said response data signals transmitted from said task module into a format that said selected said one or 5 more peripheral modules recognize.

M

39. The method as claimed in claim 38 further comprising the step of:

storing said event data signals, said one or more scripts and said response data signals in a storage medium that is in 5 communication with said task module.

40. The method as claimed in claim 39 wherein said storage medium is persistent.

41. The method as claimed in claim 34 further comprising the step of:

accessing a protocol to interface between said task module and selected said one or more peripheral modules.

Su Ba > 42. The method as claimed in claim 34 further comprising the step of:

providing communication between said task module and each of said one or more peripheral modules such that said task module will access only ones of said one or more peripheral modules capable of performing processing operations.
5

Pk > 43. The method as claimed in claim 34 wherein said executing step includes the step of:

interfacing with a plurality of said one or more peripheral modules substantially simultaneously.

44. The method as claimed in claim 34 wherein said executing step executes a plurality of said one or more scripts substantially simultaneously.

end
81°
45. The method as claimed in claim 34 further comprising the step of:

programming said one or more scripts such that said one or more instructions will be executed based upon a previously 5 executed instruction.

D
46. The method as claimed in claim 34 further comprising the step of:

providing results of said executing step to an administrative module for presenting information relating to 5 said one or more peripheral modules.

47. An event-driven data processing system comprising:

generating means for generating event data signals;

converting means for recognizing the event data signals received and mapping said event data signals to a selected set 5 of self-modifying process steps;

executing means for executing said selected set of self-modifying process steps;

processing means for providing processed data to said executing means; and

10 wherein results of said executing means are incorporated into said selected set of self-modifying process steps, thereby determining the manner in which said selected set of self-modifying process steps is executed.

48. The data processing system as claimed in claim 47 further comprising:

responding means communicating with said executing means for generating response signals based upon results of said 5 executing means.

49. The data processing system as claimed in claim 48 further comprising:

storage means for storing data generated by said generating means, said processing means, said converting 5 means, said executing means and said responding means.

50. The data processing system as claimed in claim 48 further comprising:

second transmitting means in communication with said responding means for transmitting said response signals to a 5 utilization location.

51. The system as claimed in claim 50 wherein the utilization location includes at least one processing module.

52. The system as claimed in claim 50 wherein the utilization location includes at least one event module.

53. The system as claimed in claim 47 further comprising:
administrative means for controlling said event-driven
data processing system.

54. The system as claimed in claim 47 further comprising:
managing means for dynamically modifying operating
functions of said processing means.

55. The system as claimed in claim 47 further comprising:
means for determining operational status of said
processing means.

56. The system as claimed in claim 47 further comprising:
programming means for programming said one or more sets
of self-modifying process steps.

57. The system as claimed in claim 47 further comprising:
controlling means for controlling said executing means
such that said executing means will not interface with said
processing means when said processing means are not available
5 to perform processing.

58. The system as claimed in claim 47 wherein said executing
means substantially simultaneously executes a plurality of
said one or more sets of self-modifying process steps.

59. A process for generating executable operational commands for gathering data comprising the steps of:

 collecting inputs from at least one source of information;

5 processing the inputs to generate at least one set of self-modifying commands that can be executed;

 storing said at least one set of self-modifying commands in a computer-readable medium; and

 selectively executing one or more of said at least one

10 set of self-modifying commands in response to an event data signal.

*add
B'')*